BAAN IV

BaanSync Scheduler Interface Configuration Guide

A publication of:

Baan Development B.V.
P.O.Box 143
3770 AC Barneveld
The Netherlands

Printed in the Netherlands

© Baan Development B.V. 1998.
All rights reserved.

The information in this document is subject to change without notice. No part of this document may be reproduced, stored or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Baan Development B.V.

Baan Development B.V. assumes no liability for any damages incurred, directly or indirectly, from any errors, omissions or discrepancies between the software and the information contained in this document.

Document Information

Code:	U7060C US
Group:	User Documentation
Edition:	C
Date:	April 1998

Table of contents

1 Preconditions 1-1

2 To create record in the Interface Parameters (tiint000) table 2-1

3 To create a record in the BCL Parameters (tibcl000) table 3-1

4 To create a record in the Path for Dumps (tibcl090) table 4-1

5 To import exchange schemes from the dump 5-1

Create Table from Sequential Dump (ttaad4227m000) 5-1
Maintain Exchange Schemes (tuxch0101m000) 5-3
Maintain Batches (tuxch0104m000) 5-5
Create Import Scripts (tuxch0227m000) 5-6

6 To create default data in BaanSync Scheduler 6-1

7 Condition list 7-1

About this document

This document describes the activities that must be performed after you install the BAAN IV – BaanSync Scheduler Interface. These activities are necessary to set up the interface environment for the user.

All the activities mentioned in this document are mandatory.

Chapter 1 describes the activities that you must carry out prior to the activities mentioned in this guide.

Chapter 2, 3, and 4 describe the the interface parameters which you must set.

Chapter 5 describes how you can import schemes into the new company.

Chapter 6 describes the default data that you must create in BaanSync Scheduler.

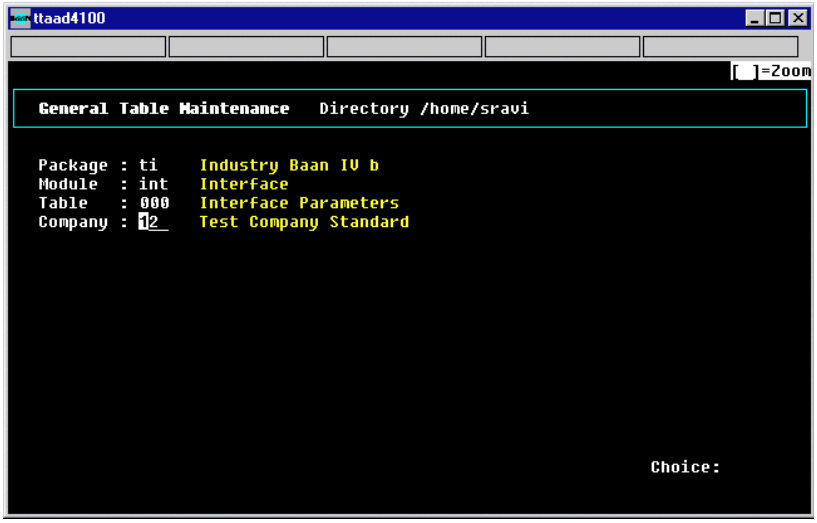
1 Preconditions

Before you proceed with the activities that are mentioned in this document, you must make sure that the following activities have been carried out successfully.

- The installation of BAAN IV*c*x
- The installation of the BAAN IV*c*x – BaanSync Scheduler VRC
- The creation of companies for the Interface
- The installation of BaanSync Scheduler 6.0 on the Windows NT system
- The proper network communication between the Windows NT system and the UNIX system
- The installation of the BaanSync Scheduler scripts that are required to export and import data from BaanSync Scheduler

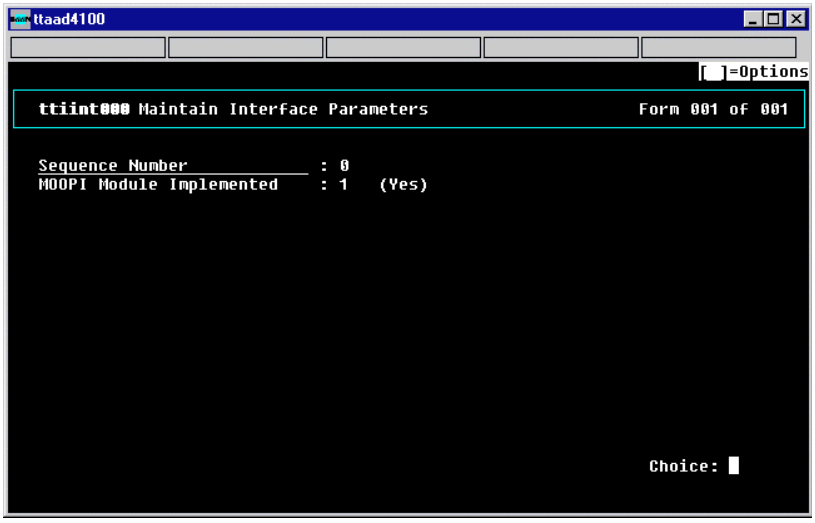
2 To create record in the Interface Parameters (tiint000) table

- Log on as a user
- Start the General table Maintenance (ttaad4100) session as shown below:



- Change the company number to that of the user.

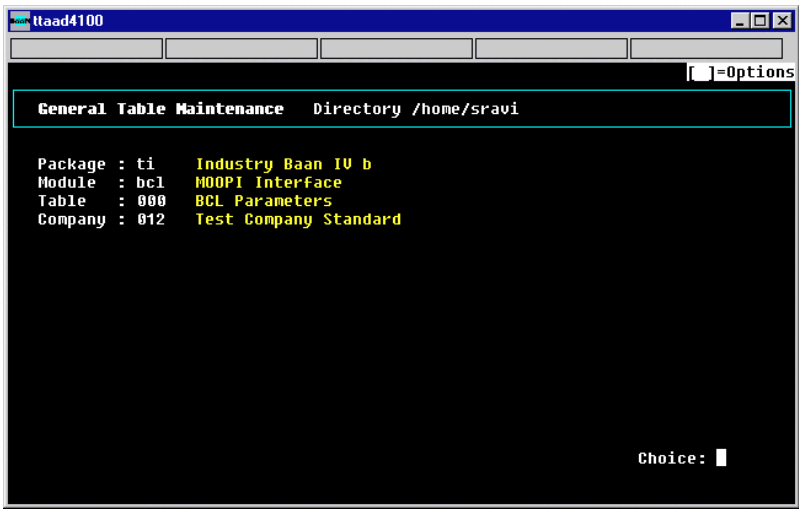
- Type Y at the prompt. The following screen is displayed:



- Type I at the prompt.
- Go to the BaanSync Scheduler Module Implemented field.
- Press spacebar.
- Select YES.
- Press enter and exit the session.

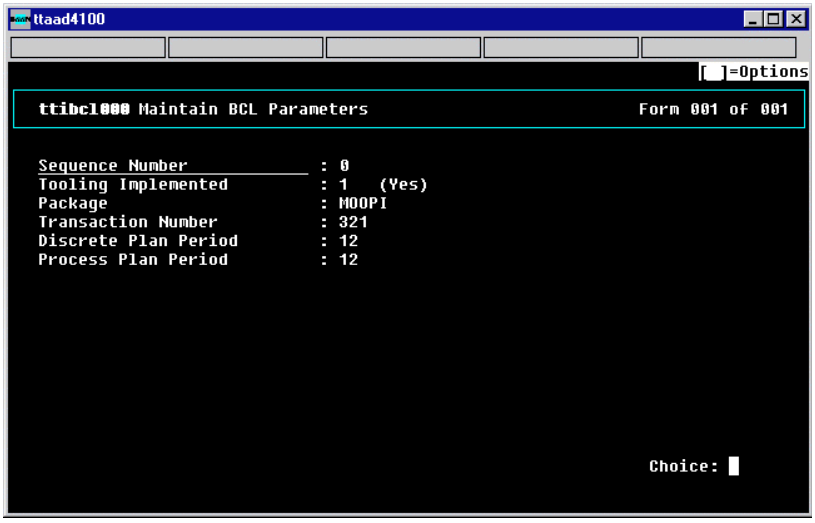
3 To create a record in the BCL Parameters (tibcl000) table

- Logon as a user.
- Start the General Table Maintenance (ttaad4100) session as shown in the following picture:


ttaad4100
[ ]=Options
General Table Maintenance Directory /home/sravi
Package : ti Industry Baan IV b
Module : bcl MOOPI Interface
Table : 000 BCL Parameters
Company : 012 Test Company Standard
Choice:

- Change the company number to that of the user.

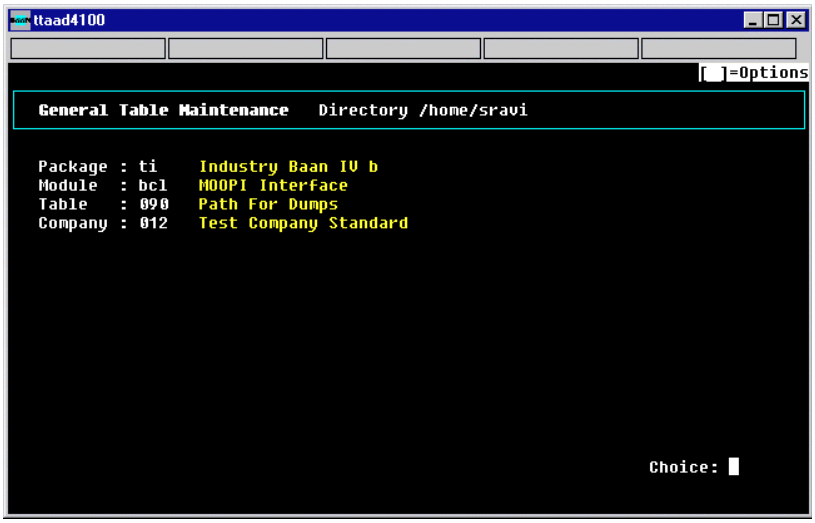
- Type Y at the prompt. The following screen will be displayed:



- Type I at the prompt.
- **Tooling Implemented** field
  - Press the spacebar
  - Select yes
- **Package** field
  - Enter **BaanSync Scheduler**
- **Discrete plan period** field
  - Enter the period in number of days
- **Process plan period** field
  - Enter period in number of days
- Press enter and exit the session.

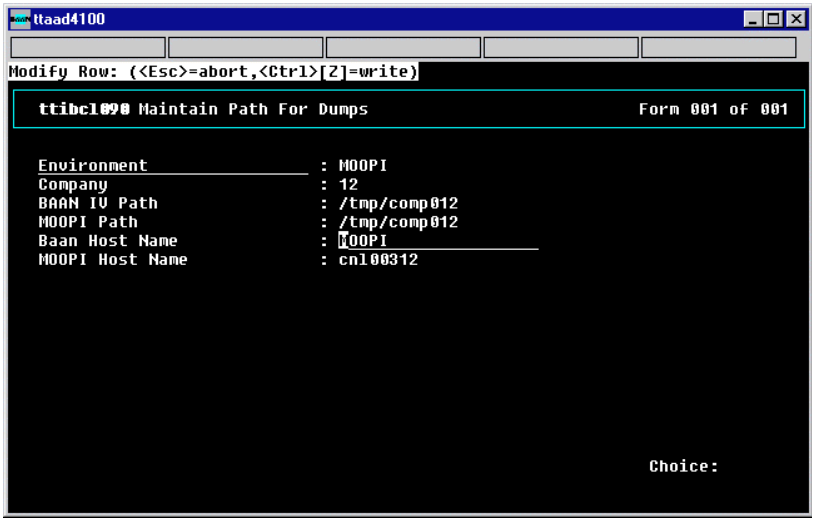
4 To create a record in the Path for Dumps (tibcl090) table

- Log on as a user.
- Start the General table Maintenance (ttaad4100) session as shown below:


ttaad4100
[ ]=Options
General Table Maintenance Directory /home/sravi
Package : ti Industry Baan IV b
Module : bcl MOOPI Interface
Table : 090 Path For Dumps
Company : 012 Test Company Standard
Choice:

- Change the company number to that of the user.

- Type Y at the prompt. The following screen is displayed:



- Type I at the prompt.
- **Environment** field
  - Type BaanSync Scheduler
- **Company** field
  - Enter the company number that is assigned to the user.
- **BAAN IV Path** field
  - Enter the directory in which you want to create the BAAN export.
  - You are advised to create the directory as is shown on the screen.
  - The directory name must always be **comp** combined with the **company** number.

NOTE: You must approach this field through the session tibcl0190m000. This session creates the directory if it does not exist.

- BaanSync Scheduler **Path** field
  - Enter the path of the directory in which the BaanSync Scheduler export files will be created.
  - You are advised to create the directory as is shown on the screen.
  - The directory name must always be **comp** combined with the **company** number.
  - You must approach this field through the Maintain Path for Dumps (tibcl0190m000). This session creates the directory if it does not exist.
- **BAAN Host Name** field
  - Enter the name of the UNIX system on which you have installed BAAN IV.
- BaanSync Scheduler **Host Name** field
  - Enter the name of the Unix/Windows NT server on which you have installed the BaanSync Scheduler software.
- Press enter and exit the session.

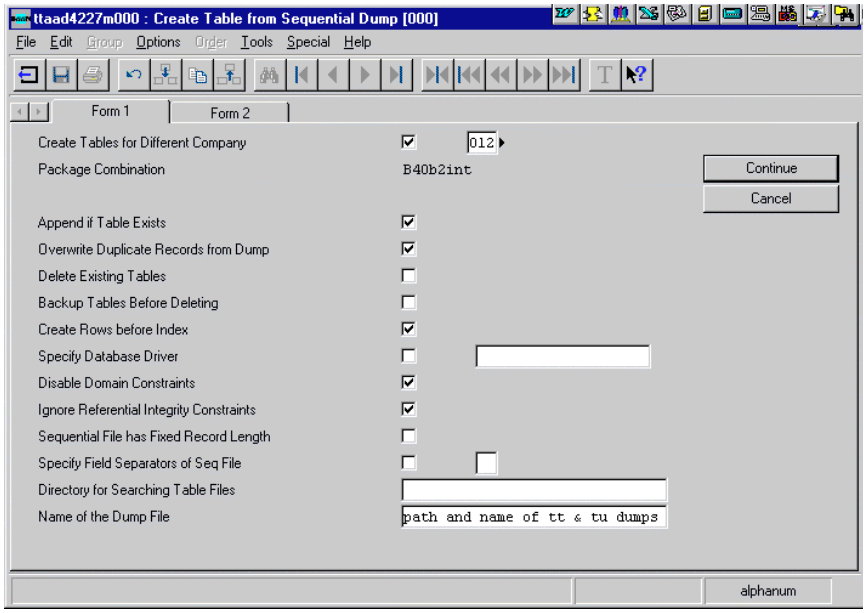
5 To import exchange schemes from the dump

This section explains the procedure to import the exchange schemes from the dump.

Create Table from Sequential Dump (ttaad4227m000)

NOTE: If you are already using the demo company supplied for the BaanSync Scheduler interface, do not implement this section.

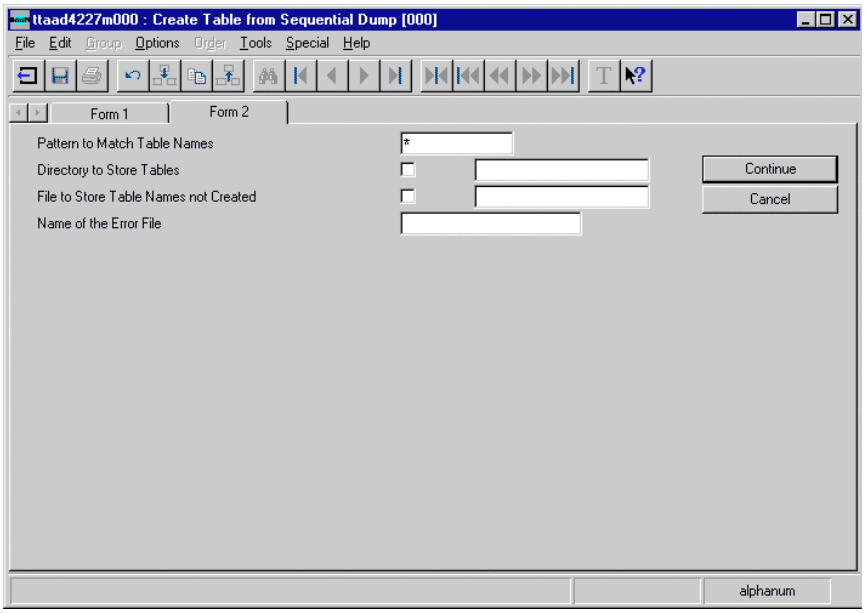
Start the session. The following screen is displayed:



Specify the options as shown above.

Specify the company number in the field next to the Create Tables for Different Company check box.

Specify the tt and tu dump file names in the Name of the Dump File field.

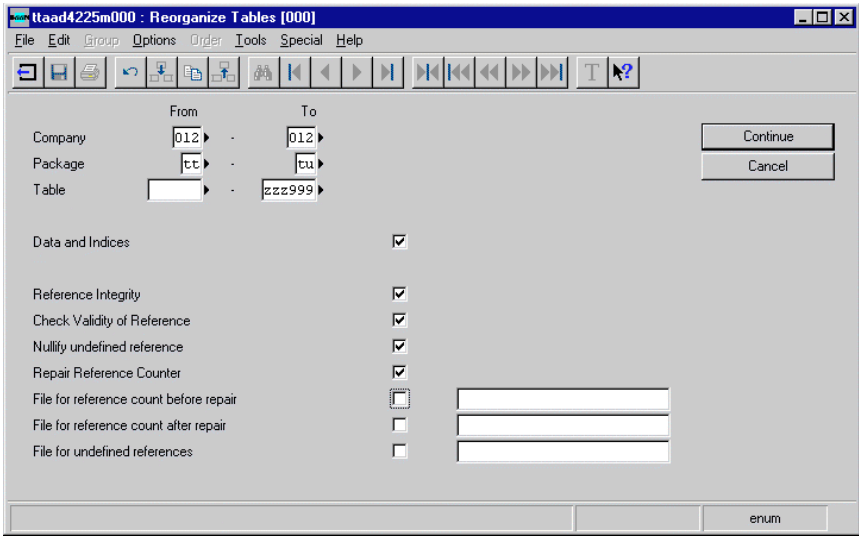


Specify the name of the error files on the second form which is shown above.

Carry out this session for both the **tt** and the **tu** dumps.

The above session will prompt to reorganize the tables. You can also reorganize tables in the Reorganize Tables (ttaad4225m000) session, which is as shown below.

Select the options as shown on the screen below:



Proceed to the next session.

Maintain Exchange Schemes (tuxch0101m000)

When you start this session, the following screen is displayed. Make sure that the exchange scheme for **moopi** is displayed.

Enter the following data:

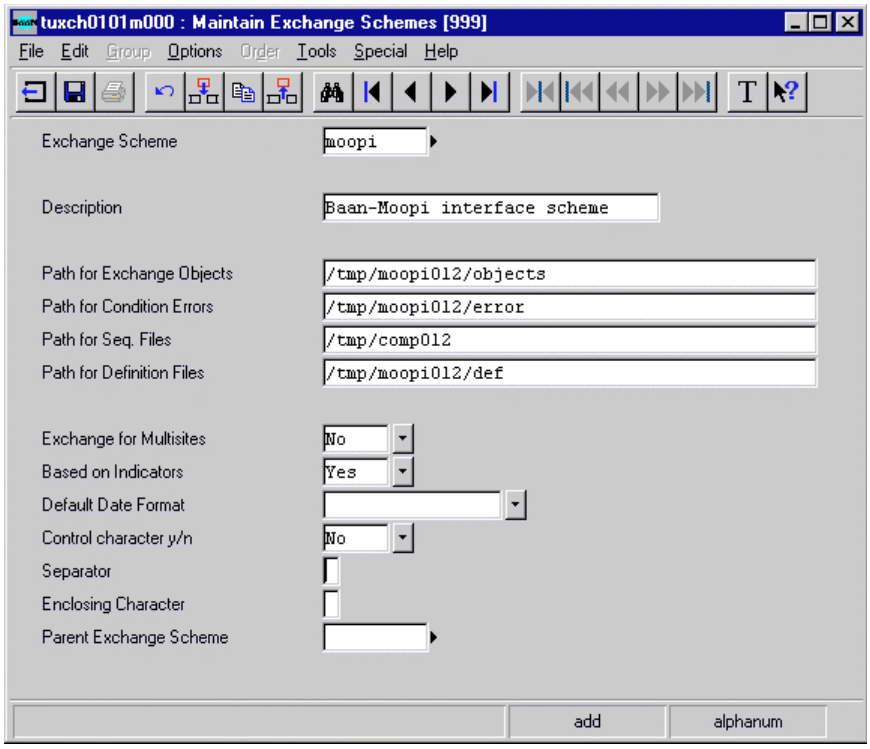
Path for Exchange Objects
The directory in which you store the objects of this company.

Path for Condition Errors
The directory in which you store the errors

Path for Sequential Files
The directory in which you store the files that are generated by BaanSync Scheduler. This path must be identical to the BaanSync Scheduler Path which you specified in the Maintain Path for Dumps (tibcl0190m000) session.

Path for Definition Files
The directory in which you store the definitions for this company.

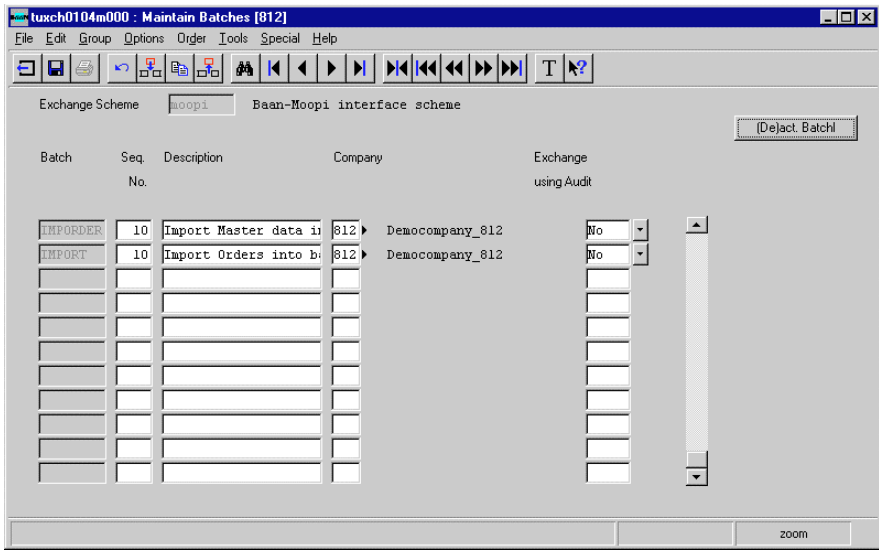
The data in the other fields should be entered as shown on the following screen:



Save and exit the session.

Maintain Batches (tuxch0104m000)

In this session, you should change the company number to the company for which the exchange scheme is maintained.

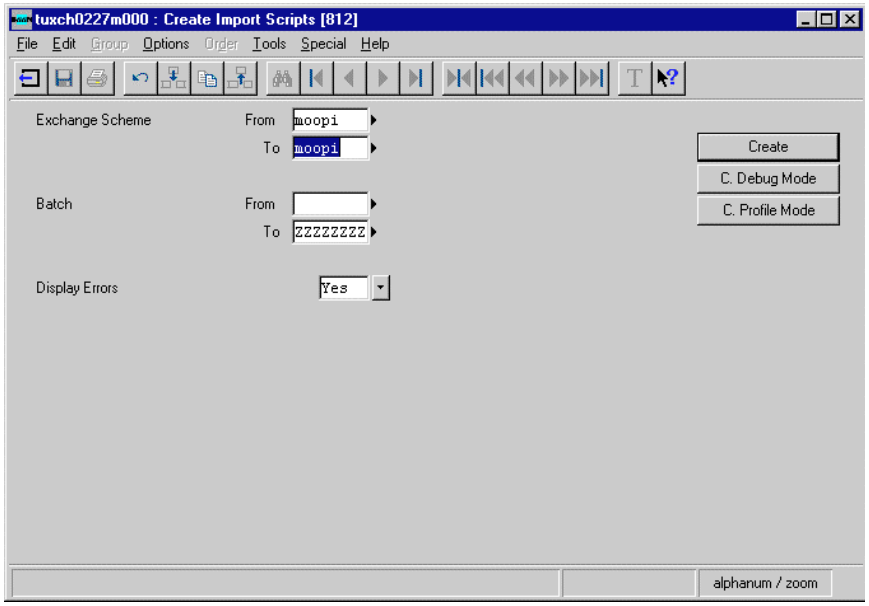


Save and proceed to the next session.

Create Import Scripts (tuxch0227m000)

Use this session to generate the import scripts for the exchange scheme that was previously defined .

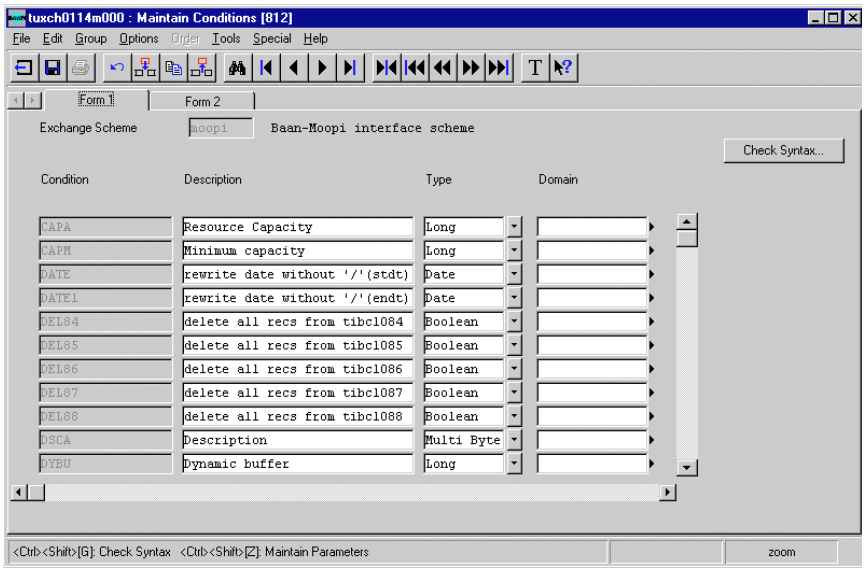
When you start this session, the following screen is displayed:



Enter **moopi** in the Exchange Scheme field and click the **Create** button or enter Y at the choice prompt.

In case of errors carry out the following procedure.

1. Find the condition code on which the error is given.
2. Refer to the documentation on conditions that is available in the user manual.
3. Edit the condition text and check if it is the same as the text that is provided in the User documentation. You can check the conditions by starting the Maintain Conditions (tuxch0114m000) session, as shown below.
4. Press CTL<G> at the choice prompt, after you modify the conditions.
5. Run the Create Import scripts (tuxch0227m000) session again to check if there are errors.


tuxch0114m000 : Maintain Conditions [812]
File Edit Group Options Order Tools Special Help
Form 1
Form 2
Exchange Scheme
moopi
Baan-Moopi interface scheme
Check Syntax...
Condition
Description
Type
Domain
CAPA
Resource Capacity
Long
CAPM
Minimum capacity
Long
DATE
rewrite date without '/'(stdt)
Date
DATE1
rewrite date without '/'(endt)
Date
DEL84
delete all recs from tibcl084
Boolean
DEL85
delete all recs from tibcl085
Boolean
DEL86
delete all recs from tibcl086
Boolean
DEL87
delete all recs from tibcl087
Boolean
DEL88
delete all recs from tibcl088
Boolean
DSCA
Description
Multi Byte
DYBU
Dynamic buffer
Long
<Ctrl><Shift>[G]: Check Syntax <Ctrl><Shift>[Z]: Maintain Parameters
zoom

6 To create default data in BaanSync Scheduler

Use the following data by default, for each database created in BaanSync Scheduler.

Units
Class: CLASS
System: SYSTEM

Customer
INV

7 Condition list

```
CAPA               Resource Capacity                Long
Condition Script    :
 if capa = "*" then
        return(0)
 endif
 return(lval(capa))

CAPM               Minimum capacity                 Long
Condition Script    :
 if capm = "*" then
        return(0)
 endif
 return(lval(capm))
```

```
DATE               rewrite date without '/'(stdt)    Date
Condition Script    :
 string day(2)
 string month(2)
 string ord.type(3)
 string year(4)
 string ret.stdt(10)
 domain tuxch.casf file
 domain tcpdno     ord.no
 domain tcorno     oprn.no
 domain tcdate     ret.date

 file = ascii.file
 string.scan(opno,"%s/%d/%d", ord.type,ord.no,oprn.no)
 if stdt = "*"  then
        |* In tibcl084, the start date should not be empty
        if file = "tibcl084" then
                log.mess("tibcls0157", pdno, oprn.no)
                |* Start date for order no %s and operation %s is
NULL
                errors = 2
else
ret.date = 0                                  endif
 else
        string.scan(stdt, "%s/%s/%s", day,month,year)
        ret.stdt = day & month & year
        ret.date = inputstr.to.date(ret.stdt, "%D002,2")
 endif
 return(ret.date)
```

```
DATE1              rewrite date without '/'(endt)    Date
Condition Script    :
 string day(2)
 string month(2)
 string ord.type(3)
 string year(4)
 string ret.endt(10)
 domain tuxch.casf file
 domain tcpdno     ord.no
 domain tcorno     oprn.no
 domain tcdate     ret.date

 file = ascii.file
 string.scan(opno, "%s/%d/%d", ord.type, ord.no, oprn.no)
 if endt = "*"  then
        |* In tibcl084, the end date should not be empty
        if file = "tibcl084" then
                log.mess("tibcls0158", pdno, oprn.no)
                |* End date for order no %s and operation %s is NULL
                errors = 2
        else
                ret.date = 0
        endif
 else
        string.scan(endt, "%s/%s/%s", day,month,year)
        ret.endt = day & month & year
        ret.date = inputstr.to.date(ret.endt, "%D002,2")
 endif
 return(ret.date)
```

```
DEL84              delete all recs from tibcl084     Boolean
Condition Script    :
 table  ttibcl084
 Detailed Planning Production

long   scan.ret

 scan.ret = db.clear.table(ttibcl084,1)
 if scan.ret = 0 then
        return(true)
 else
        log.mess("tibcls0155","tibcl084")
        |* Error while deleting records from %s
        return(false)    |* Stops execution
 endif
```

```
DEL85              delete all recs from tibcl085     Boolean
Condition Script    : | table  ttibcl085       | Detailed Planning
Employees
 long   scan.ret

 scan.ret = db.clear.table(ttibcl085,1)
 if scan.ret = 0 then
        return(true)
 else
        log.mess("tibcls0155","tibcl085")
        |* Error while deleting records from %s
        return(false)    |* Stops execution
 endif
```

```
DEL86              delete all recs from tibcl086     Boolean
Condition Script    :
 table  ttibcl086       | Detailed Planning Tools
 long   scan.ret

 scan.ret = db.clear.table(ttibcl086,1)
 if scan.ret = 0 then
        return(true)
 else
        log.mess("tibcls0155","tibcl086")
        |* Error while deleting records from %s
        return(false)    |* Stops execution
 endif
```

```
DEL87              delete all recs from tibcl087     Boolean
Condition Script    :
 table  ttibcl087       | Detailed Planning Setups
long   scan.ret

 scan.ret = db.clear.table(ttibcl087,1)
 if scan.ret = 0 then
        return(true)
 else
        log.mess("tibcls0155","tibcl087")
        |* Error while deleting records from %s
        return(false)    |* Stops execution
 endif

DEL88              delete all recs from tibcl088     Boolean
Condition Script    :
 table  ttibcl088       | Purchase Recommendations
 long   scan.ret

 scan.ret = db.clear.table(ttibcl088,1)
 if scan.ret = 0 then
        return(true)
 else
        log.mess("tibcls0155", "tibcl088")
        |* Error while deleting records from %s
        return(false)    |* Stops execution
 endif

DSCA               Description                       Multi Byte
Condition Script    :
 if dsca = "*" then
        return("")
 endif
 return(dsca)

DYBU               Dynamic buffer                    Long
Condition Script    :
 tibcl013.dytu = ltoe(3)
 if dybu = "*" then
        return(0)
 endif
 return(lval(dybu) / 3600)
```

```
EMNM               Employee Model                    String
Condition Script    :
 if emnm = "*" then
        return("")
 endif
 return(emnm)

EMNO               Employee                          Long
Condition Script    :
 if emno = "*" then
        return(0)

 endif
 return(lval(emno))

ERR                Log message if error occurs       Boolean
Condition Script    :
 ^long  errors     |    Indicate if error has occurred

 ^long  err.mess

 domain tuxch.casf      file    | Name of the current ascii file
 domain tuxch.path      path    | Path of the current ascii file
being read
 file = ascii.file
 path = ascii.file.name

 if no.err.rejects > 0 or
    errors > 0 then
        log.mess("tibcls0154",path)
        |* Error while reading %s
        err.mess = 1
 endif
 if file = "tibcl022" then
        if err.mess = 1 then
                shell("echo Error while importing master data ", 4)
        else
                shell("echo Import of master data successful", 4)
        endif
 endif
 return(false)
```

```
FOLE               Following indicator               Enumerated
tibcl.fole
Condition Script    :
 if fole = "*" then
        return(ltoe(0))
 endif
 return(ltoe(lval(fole) + 1))


GRAD               Gradation Weight                  Double
Condition Script    : | if grad = "*" then
        return(0.0)
 endif
 return(val(grad))

MCNM               Machine Model                     String
Condition Script    :
 if mcnm = "*" then
        return("")
 endif
return(mcnm)

MCNO               machine no                        String

Condition Script    : | domain tirou.mcno      bcl84.mcno      fixed
 if mcno = "*" then
        return("")
 endif
 bcl84.mcno = shiftr$(mcno)
 return(bcl84.mcno)

MDUR               duration                          Long
Condition Script    : | tibcl084.mdun = ltoe(3)
 if mdur = "*" then
        return(0)
 else
        return(val(mdur) / 3600 )
 endif
```

```
NAME               Resource Model Name               Multi Byte
Condition Script    :
 if name = "*" then
        name = ""
 endif
 return(name)

NUMB               No of Employees                   Integer
Condition Script    :
| if numb = "*" then
        return(0)
 else
        return(lval(numb))
 endif

OPNO               Decoding for operation code       Integer
Condition Script    :
 string order.type(3)
 domain tcpdno  order.no
 string operation.no(3)

 string.scan(opno, "%s/%d/%s", order.type, order.no, operation.no)
 if order.type = "SFC" or
    order.type = "PMG" then
        return(val(operation.no))
 endif
 return(0)

PDNO84             Production order without sern     String
Condition Script    : | string ordr.type(3)
 string ordr.mthd(3)
 domain tcopno  serial.no
 string  prod.no(6)      fixed

 serial.no = 0
 string.scan(pdno, "%s/%s/%s/%d", ordr.type, ordr.mthd, prod.no,
serial.no)
 tibcl084.sern = serial.no
 pdno = ordr.type & "/" & ordr.mthd & "/" & prod.no
 return(pdno)
```

```
PDNO85              Production order without sern     String
Condition Script    : | string ordr.type(3)
 string ordr.mthd(3)

 domain tcopno  serial.no
 string  prod.no(6)     fixed

 serial.no = 0
 string.scan(pdno, "%s/%s/%s/%d", ordr.type, ordr.mthd, prod.no,
serial.no)
 tibcl085.srno = serial.no
 pdno = ordr.type & "/" & ordr.mthd & "/" & prod.no
 return(pdno)
```

```
PDNO86              Production order without sern     String
Condition Script    :
 string ordr.type(3)
 string ordr.mthd(3)
 domain tcopno  serial.no
 string prod.no(6)      fixed

 serial.no = 0
 string.scan(pdno, "%s/%s/%s/%d", ordr.type, ordr.mthd, prod.no,
serial.no)
 tibcl086.sern = serial.no
 pdno = ordr.type & "/" & ordr.mthd & "/" & prod.no
 return(pdno)
```

```
PODT               Planned order date                Date
Condition Script    :
 string day(2)
 string month(2)
 string year(4)
 string ret.podt(10)
 domain tuxch.casf file
 domain tcdate ret.date

 file = ascii.file
 if podt = "*"  then
        |* In tibcl084, the end date should not be empty
        if file = "tibcl084" then
 |              log.mess("tibcls0158", pdno, opno)
                |* End date for order no %s and operation %s is NULL
                errors = 2
        else
                ret.date = 0
        endif
 else
        string.scan(podt, "%s/%s/%s", day,month,year)
        ret.podt = day & month & year
        ret.date = inputstr.to.date(ret.podt, "%D002,2")
 endif
 return(ret.date)
QUAN               quantity                          Double
Condition Script    :
 if quan = "*" then
        return(0.0)
 else
        return(val(quan))
 endif


SETP               Setup Weight                      Double
Condition Script    :
 if setp = "*" then
        return(0.0)
 endif
 return(val(setp))
```

```
STAT                team status                         Enumerated
tibcl.tmtp
Condition Script    :
if stat = "*" then
        return(ltoe(0))
 else
        return(ltoe(lval(stat) + 1))
 endif

TEAM                                                    Long
Condition Script    :
 if team = "*" then
        return(0)
 else
        return(lval(team))
 endif

TIME                rewrite time without ':'(sttm)      Long
Condition Script    :
 string hr(2)
 string mm(2)
 string time1(4)

 if sttm = "*" then
        time1 = "0"
 else
        string.scan(sttm,"%s:%s", hr,mm)
        time1 = hr & mm
 endif
 return(lval(time1))

TIME1               rewrite time without ':'(entm)      Long
Condition Script    :
 string hr(2)
 string mm(2)
 string time1(4)

 if entm = "*" then
        time1 = "0"
 else
        string.scan(entm,"%s:%s", hr,mm)
        time1 = hr & mm
 endif
 return(lval(time1))
```

```
TIML               Time Losses Weight                Double
Condition Script    :
 if timl = "*" then
        return(0.0)
 endif
 return(val(timl))

TMTP               Team Type                         Enumerated
tibcl.tmtp
Condition Script    :
 on case tmtp
 case "0" :
            return(ltoe(2))
            break
 case "1" :
            return(ltoe(2))
            break
 case "2" :
            return(ltoe(1))
            break
 endcase
 return(ltoe(0))



TOLM               Tool Model                        String
Condition Script    :
 if tolm = "*" then
        return("")
 else
        return(tolm)
 endif

TOOL               Tool                              String
Condition Script    :
 if tool = "*" then
        tool = ""
 endif
 return(tool)
```

```
UPDPROD            update prod order tables          Boolean
Condition Script    :
 domain tuxch.path      path    | Path of current ascii file being
read

 path = ascii.file.name
 if no.err.rejects > 0
    or errors > 0 then
        log.mess("tibcls0154", path)
        |* Error while reading %s
        err.mess = 1
 endif

 shell("ba6.1 otibcl02851",0)
 return(false)
```

```
UPDPUR             update purchase order tables      Boolean
Condition Script    :
 domain tuxch.path path         | Path of the ascii file being read

 path = ascii.file.name
 if no.err.rejects > 0
    or errors > 0 then
        log.mess("tibcls0154", path)
        |* Error while reading %s
        err.mess = 1
 endif
 shell("ba6.1 otibcl02852", 0)
 if err.mess = 1 then
        shell("echo Error while importing scheduled order data ", 4)
 else
        shell("echo Import of scheduled order data successful", 4)
 endif
 return(false)
```

```
UTIL               Tool Utilisation                  Long
Condition Script    :
 tibcl014.uttu = ltoe(3)
 if util = "*" then
        return(0)
 endif
 return(lval(util) / 3600 )
```

```
WOCD               work order code                    Multi Byte
Condition Script    :
 if wocd = "*" then
        return("")
 else
        return(shiftl$(wocd))
 endif
```